

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 11-133945

(43)Date of publication of application : 21.05.1999

(51)Int.Cl.

G09G 5/36
G06F 3/153
G06F 13/16
G06T 11/00
G09G 5/18

(21)Application number : 09-298178

(71)Applicant : HITACHI LTD

(22)Date of filing : 30.10.1997

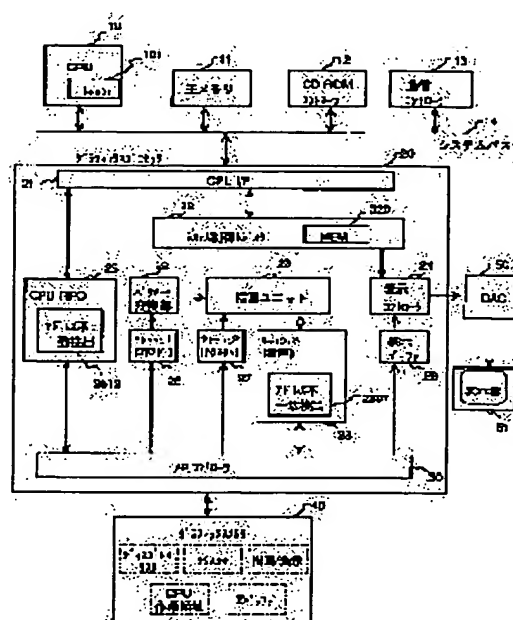
(72)Inventor : JO MANABU
MATSUO SHIGERU
NARITA MASAHIKA

(54) GRAPHICS DISPLAY DEVICE

(57)Abstract:

PROBLEM TO BE SOLVED: To reduce the buffer size necessary for buffering of display area data by controlling the lead-out timing of the display data by other memory access except for display with a graphics processor.

SOLUTION: A CPU 10 controls the whole of a device, and executes a program displayed with a figure in a display unit 51. A CD-ROM controller 12 is accessed to a CD-ROM storing receives the figure information, and a communication controller 13 transmits/receives the information between other device. A graphics processor 20 draws a figure in a display area inside of a graphics memory 40, and displays the drawn data in the display unit 51. The graphics processor 20 is provided with an address detecting means for detecting whether an upper leveled plural bits of an address to be continuously transmitted into the graphics memory 40 is intermittent or not, and in the case of transferring the data to the graphic memory 40, the number of data transference to the memory is changed on the basis of the information of the detected intermittent address.



LEGAL STATUS

[Date of request for examination] 27.03.2001

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number] 3454113

[Date of registration] 25.07.2003

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平11-133945

(43) 公開日 平成11年(1999) 5月21日

(51) Int.Cl. ⁶	識別記号	F I	
G 0 9 G 5/36	5 3 0	G 0 9 G 5/36	5 3 0 G 5 3 0 C 5 3 0 E 3 3 0 A 5 2 0 A
G 0 6 F 3/153 13/16	3 3 0 5 2 0	G 0 6 F 3/153 13/16	
審査請求 未請求 請求項の数 4 O L (全 15 頁) 最終頁に続く			

(21) 出願番号 特願平9-298178

(22) 出願日 平成9年(1997)10月30日

(71) 出願人 000005108

株式会社日立製作所

東京都千代田区神田駿河台四丁目6番地

(72) 発明者 城 学

茨城県日立市大みか町七丁目1番1号 株式会社日立製作所日立研究所内

(72) 発明者 松尾 茂

茨城県日立市大みか町七丁目1番1号 株式会社日立製作所日立研究所内

(72) 発明者 成田 正久

茨城県日立市幸町三丁目2番1号 日立エンジニアリング株式会社内

(74) 代理人 弁理士 小川 勝男

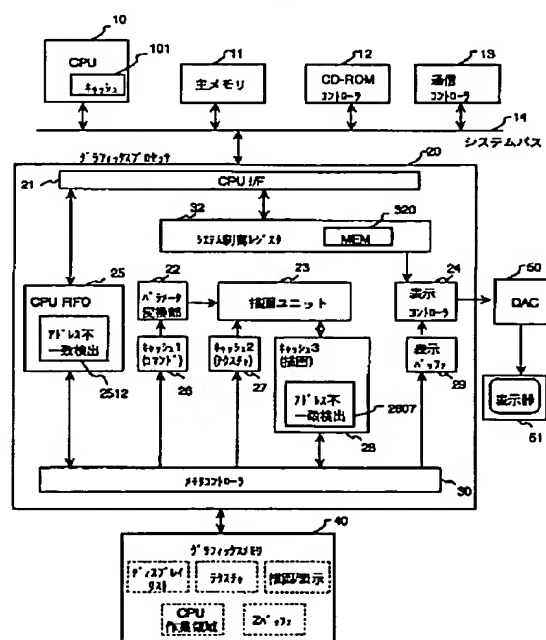
(54) 【発明の名称】 グラフィックス表示装置

(57) 【要約】

【課題】 CPUとグラフィックスプロセッサが共通のグラフィックスメモリをアクセスするメモリ統合型のグラフィックス表示装置において、常に連続したアドレスでメモリにアクセスすることにより効率の良いメモリアクセスを実現するグラフィックスプロセッサを得る。

【解決手段】 上記目的を達成するため、本発明では、グラフィックスプロセッサに対してグラフィックスメモリに書き込むデータのアドレスの不連続を検出する手段を設け、不連続の場合は書き込み時間が長くなると判断し、一度書き込み要求を解除する。

図 1



【特許請求の範囲】

【請求項 1】表示すべきグラフィックス図形の種類や頂点パラメータ等で構成される描画手続き情報を生成する CPU と、前記描画手続き情報やビットマップ情報を記憶するメモリと、前記メモリ上の描画データを表示する表示器と、前記描画手続き情報に基づいて前記メモリ上に順次図形を描画し、さらに前記ビットマップ情報を前記表示器に表示するために前記メモリの表示読み出しを行うグラフィックスプロセッサを備え、前記メモリは、前記 CPU と、前記グラフィックスプロセッサの両方がアクセスするグラフィックス表示装置であって、前記グラフィックスプロセッサは前記メモリに連続して転送するアドレスの上位複数ビットが不連続かどうかを検出するアドレス検出手段を備え、前記グラフィックスプロセッサが前記メモリへデータを転送する場合、前記不連続アドレス検出手段で検出したアドレス不連続情報によって前記メモリに対するデータの転送数を変化させることを特徴とするグラフィックス表示装置。

【請求項 2】請求項 1 に記載のグラフィックスプロセッサは、前記アドレス不連続情報が不連続を示している場合、前記グラフィックスプロセッサから前記メモリへの転送要求を一度解除することによって無駄な調停時間を省き、メモリアクセスの時間を短縮できることを特徴とするグラフィックス表示装置。

【請求項 3】請求項 1 に記載のグラフィックスプロセッサは、前記不連続アドレス検出手段でアドレスを比較する場合、前記メモリの種類によって比較するビットの幅を指定する情報を持ち、前記ビット幅情報によって比較するビット幅を変化させることを特徴とするグラフィックス表示装置。

【請求項 4】少なくともメモリに対してビットマップ情報を生成するための描画アクセス、表示器に表示データを出力するための表示アクセスを持ち、各々のアクセスに対して連続したデータ転送を行うグラフィックスプロセッサであって、前記グラフィックスプロセッサはデータ転送中のアドレスの一部が不連続であることを検出する不連続アドレス検出手段を備え、前記アクセス内容のいずれかのアクセスを行っている時に前記不連続アドレス検出手段が不連続を検出した場合、当該アクセス以外のアクセスに切り替えることを特徴とするグラフィックスプロセッサ。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は図形表示装置に係り、特に小型、低価格システムでグラフィックス動画表示を可能とするため複数の情報を同一のメモリに統合し

たグラフィックス表示装置に関する。

【0002】

【従来の技術】高速な三次元グラフィックスを処理するプロセッサの例として、“3次元CG描画LSI、パソコンで30万ポリゴン/秒実現：日経エレクトロニクス、1995年7月17日(No. 640)、pp109-120”が紹介されている。このプロセッサはプロセッサ専用メモリとしてテキストチャ用メモリ、フレームバッファ用メモリ、ローカル用メモリの3種類を設けている。このアーキテクチャは性能向上を図る上で有利であるが、メモリが複数通り必要なため個人向け携帯用機器等の小型、低価格な装置には不向きである。そこで、CPUの主メモリにグラフィックス情報を一元化し、メモリ個数を削減したグラフィックスシステムの例として、特開平5-257793号がある。これは1種類のメモリの中に、CPUのプログラム、テキストチャデータ、フレームバッファ等を統合して持つシステムが記載されている。

【0003】

【発明が解決しようとする課題】先の従来技術によれば、メモリアクセス能力が数百MB/sといった十分高速であることが前提となっている。従って先の従来技術においては、表示データの読み出しの時間は十分確保されている。高速なメモリシステムを持つためには、メモリアクセスのデータバス幅を広くするか、高速なメモリを持つことになり、このことはシステムの小型化、低価格化の妨げになる。

【0004】そこでメモリアクセス能力を下げると、表示データの読み出し時間の確保のために、表示以外の他のメモリアクセスによって表示データの読み出しタイミングの制御が必要となる。特にCPUからのアクセスは頻度が高いので、表示データの読み出しに影響する。先の従来技術に於いては表示以外の他のメモリアクセスによる表示データの読み出しタイミングの制御について述べられていない。

【0005】そこで本発明の目的は、表示以外の他のメモリアクセスによる表示データの読み出しタイミングの制御をグラフィックスプロセッサが行うことによって、表示領域データのバッファリングを行う為に必要なバッファサイズを小さくできるグラフィックス表示システム及び方法を提供することにある。

【0006】

【課題を解決するための手段】上記目的を達成するため、本発明では、表示すべきグラフィックス図形の種類や頂点パラメータ等で構成される描画手続き情報を生成するCPUと、描画手続き情報やビットマップ情報を記憶するメモリと、メモリ上の描画データを表示する表示器と、描画手続き情報に基づいてメモリ上に順次図形を描画し、さらにビットマップ情報を表示器に表示するためにメモリの表示読み出しを行うグラフィックスプロセッサを備え、メモリは、CPUと、グラフィックスプロセ

ッサの両方がアクセスするグラフィックス表示装置であって、グラフィックスプロセッサはメモリに連続して転送するアドレスの上位複数ビットが不連続かどうかを検出するアドレス検出手段を備え、グラフィックプロセッサがメモリへデータを転送する場合、不連続アドレス検出手段で検出したアドレス不連続情報によってメモリに対するデータの転送数を変化させることを特徴とする。

【0007】

【発明の実施の形態】図1は、本発明に係るグラフィックスプロセッサを用いた図形処理装置のシステム構成例を示す。CPU10は装置全体の制御を行うと共に、表示器51に図形を表示するためのプログラムを実行する。CPU10は内部にキャッシュ101を内蔵する。主メモリ11は、CPU10が処理するデータやプログラムを記憶するメモリである。CD-ROMコントローラ12は、図形情報を記憶しているCD-ROMをアクセスするためのコントローラであり、通信コントローラ13は図示しない他の装置との間で情報を送受するためのコントローラである。グラフィックスプロセッサ20は、グラフィックスメモリ40内の表示領域に図形を描画し、さらに描画したデータを読み出して表示器51に図形を表示するためのプロセッサである。DAC (Digital to Analog Converter) 50は、グラフィックスプロセッサ20が出力するデジタル形式の表示データをアナログデータに変換する。

【0008】また、グラフィックスメモリ40を構成する素子としては、DRAM(Dynamic RAM)を用いるのが望ましい。これは、DRAMが他のメモリに比べて、チップ面積に対するトランジスタの集積度が高いためである。DRAMは不連続なアドレスでアクセスするとアクセス時間が長い。しかし、DRAMは高速ページモードアクセスというアクセス方式を持っており、アドレスの上位部分(例えばビット9以上)が一致している場合の連続アクセスは高速アクセスが可能になる特徴を持っている。

【0009】本図形処理装置で表示する図形は、グラフィックス動画表示を行うことを目的とする。つまり、1/60秒や1/30秒単位で図形の大きさや位置を少しずつ変化させ、画面を連続的に見ることでグラフィックス図形の動画表示を行うものである。従って、CPU10やグラフィックスプロセッサ20は、1/60秒や1/30秒毎に1画面分の描画を行わなければならない。1画面分の図形を描画するためには、次のような手順となる。

【0010】(1) CPU10による図形データの座標変換

表示すべき図形に対して、方向や大きさ等を計算し、図形の頂点座標の計算を行う。一般に複雑な図形は、三角形や四角形といった単純な図形を多数組み合わせで構成される。従って、これらの図形の全ての頂点座標を計算

する。

【0011】(2) CPU10によるディスプレイリストの作成

グラフィックスプロセッサ20を用いて前記の多数の単純図形をグラフィックスメモリ40に描画するために、グラフィックスプロセッサ20が実行できるコマンドの形式に変換してグラフィックスメモリ40に転送する。通常、前記多数の単純図形の分だけコマンドが連結される。このコマンドが連結したものをディスプレイリストと呼ぶ。ディスプレイリストは数10から数100Kバイトの大きさとなる。

【0012】(3) グラフィックスプロセッサ20による描画

前記のディスプレイリストをグラフィックスプロセッサ20が順次読み込み、そのリストに示されるコマンドに従ってグラフィックスメモリ40内の描画領域に描画する。

【0013】(4) グラフィックスプロセッサ20による表示

前記の描画された図形は、グラフィックスプロセッサ20によって表示器51に合わせたタイミングで読み出されて、表示器51に表示される。

【0014】以上の(1)から(4)を1/60秒や1/30秒毎に繰り返す。

【0015】次に、グラフィックスプロセッサ20の内部の概略について説明する。CPU I/F 21は、CPU10がシステム制御レジスタ32等のレジスタ類やグラフィックスメモリ40をアクセスするための制御を行う。描画ユニット23は、グラフィックスメモリ40内のディスプレイリストをフェッチし、そのリストに示されるコマンドに従って描画を行う。パラメータ変換部22は、必要に応じて前記コマンドのパラメータを変換する。表示コントローラ24は、描画ユニット23が描画したデータを表示するための制御部である。上記のようにグラフィックスプロセッサ20は、何らかの処理を行う毎にグラフィックスメモリ40をアクセスするため、グラフィックスメモリ40のアクセス効率を高めることが処理速度の向上につながる。そこでグラフィックスプロセッサ20は、各アクセス要求単位にキャッシュやFIFOを持つことでアクセス効率を高めている。CPU FIFO 25は、CPU10のグラフィックスメモリ40のアクセスを高速化する。アドレス不一致検出2512はグラフィックスメモリ40に書き込むデータのアドレスが不連続かを検出する。キャッシュ116はコマンド専用、キャッシュ227はテクスチャ専用、キャッシュ328は描画専用である。アドレス不一致検出2807はグラフィックスメモリ40に書き込むデータのアドレスが不連続かを検出する。また、表示データ用に表示バッファ29を持つ。メモリコントローラ30は、前記キャッシュやFIFO等のグラフィックスメモ

リ 40 に対するアクセス要求を受付て優先順位を決定し、前記メモリのアクセスを制御する。メモリコントローラ 30 は、表示コントローラ 24 からのアクセス要求を最も優先するが、CPU 10 や描画ユニット 23 からのアクセスが行われている間は、それらは中断されず表示コントローラ 24 が待たされる場合がある。システム制御レジスタ 32 は、グラフィックスプロセッサ 20 の動作モードを指定するレジスタである。このレジスタの中にはアドレス不一致検出 2512、2807 で比較するアドレスの上位複数ビットの幅を指定する MEM (Memory Mode) ビットがある。例えば 4M ビットの DRAM を使う場合は上位 13 ビット、16M ビットの DRAM を使う場合は上位 12 ビットを比較するように指定する。

【0016】次に、図 2 においてグラフィックスプロセッサ 20 の端子機能について説明する。

【0017】(1) System 系

システムモードの設定とクロック及びリセット入力する端子である。グラフィックスプロセッサ 20 は、描画系と表示系で独立したクロックを入力することができる。従って、表示器 51 の種類に関係なく、描画系は常に高速処理を行うことができる。

【0018】(2) CPU 系

CPU I/F 用の端子である。CPU 10 は、グラフィックスメモリ 40 の全空間と、前記システム制御レジスタ 32 等の内部のレジスタをアクセスすることができる。グラフィックスメモリ 40 をアクセスする場合は CS0 端子を Low に、レジスタをアクセスする場合は CS1 端子を Low にする。グラフィックスメモリ 40 へのライトアクセスはバイト単位が可能となるようにライトインエーブルを 2 本持つ。このほか、DMA 転送を制御する DREQ、DACK 端子や、バスサイクルを延長する WAIT 端子、CPU 10 に対する割り込みを発生させる IRL 端子がある。

【0019】(3) Power 系

電源を供給する端子は、クロック制御を行う PLL 専用の端子と、その他の一般用がある。

【0020】(4) Display 系

表示用の端子として、ドットクロック出力 (DCLK)、表示データ出力 (DD0-DD15)、同期信号の入出力端子 (HSYNC、VSYNC) 等がある。

【0021】(5) Memory 系

グラフィックスメモリ 40 との I/F として、DRAM (Dynamic Random Access Memory) を直結できる端子を持つ。

【0022】次に、図 3 においてグラフィックスプロセッサ 20 の描画コマンドを説明する。

【0023】(1) 四角形描画コマンド

矩形のテクスチャデータを任意の四角形に変形させながら描画する。テクスチャデータが 2 値の場合はカラー拡

張を行う。

【0024】(2) LINE

単一の直線、または複数の直線を描画する。

【0025】(3) MOVE

描画開始点の移動を行う。

【0026】(4) LOFS

描画座標の原点をずらすコマンドである。このコマンドが実行された後のコマンドはディスプレイリストに示される座標パラメータに対してこのコマンドで指定した分だけ座標をずらして描画する。

【0027】(5) AFFIN

図形を描画する場合に、回転や拡大、縮小を指定するコマンドである。ディスプレイリストに示される座標パラメータに対してこのコマンドで指定した分だけ座標を回転 (または拡大、縮小) して描画する。

【0028】(6) JUMP

ディスプレイリストを分岐させるコマンドである。

【0029】(7) GOSUB

ディスプレイリストのサブルーチンをコールする。

【0030】(8) RET

サブルーチンから復帰する。

【0031】(9) TRAP

ディスプレイリストのフェッチを終了する。

【0032】(10) FLASH

テクスチャデータのキャッシュであるキャッシュ 227 内に存在するデータを無効化し、新たにグラフィックスメモリ 40 からデータを読み込ませる。

【0033】次に、図 4 においてグラフィックスプロセッサ 20 のレジスタ機能について説明する。

【0034】(1) システム制御レジスタ

SRES は描画ユニット 23 をソフトウェアによって初期化し、DRES は表示コントローラ 24 をソフトウェアによって初期化する。DAC は、表示領域 (フレームバッファ領域) を切り替える。RS は、ディスプレイリストのフェッチを開始させる。CAM は CPU 10 内のキャッシュ 101 の種類を指定する。CPU 10 がグラフィックスメモリ 40 にデータをストアする時の CPU 10 の動作の特徴として、CPU 10 のキャッシュ 101 がコピーバック方式を採用している場合はキャッシュのラインサイズ分だけまとめてデータをライトするが、ライトスルー方式を採用している場合は 1 ワード単位でデータをライトする。

【0035】ここでアドレスの連続性によるグラフィックスメモリ 40 への書き込み時間の違いについて説明する。

【0036】前記 CPU FIFO25 からグラフィックスメモリ 40 にデータを書き込む場合、書き込まれるデータのアドレスが連続している場合は、先に DRAM の特徴のところで述べたように CPU FIFO25 から短い時間でグラフィックスメモリ 40 へ書き込むことができる。この場合 1

6ワードのデータをライトすると、約20サイクルでデータの書き込みができる。一方、書き込まれるデータのアドレスが不連続の場合、先に述べたようにDRAMの特性によりCPU FIFO25からグラフィックスメモリ40に書き込む時間が長くなる。前記CPU FIFO25には、最悪の場合は全てが不連続な場合もありうる。この場合は、アドレスが連続している場合に比べ4倍くらいの時間（最大80サイクル）がかかる可能性がある。そこでグラフィックスメモリ40にデータを書き込む場合、常に連続したアドレスにすることによって書き込み時間を短縮することができ、効率的なメモリアクセスが可能となる。この動作については後で詳細に説明する。

【0037】(2) ステータスレジスタ

VBKは、表示のフレーム切り替えを通知する。TRAは、TRAPコマンドを実行しディスプレイリストのフェッチを終了したことを通知する。DBFは、2つのフレームバッファに対してどちらを現在表示中かを示す。

【0038】(3) ステータスレジスタ・クリアレジスタ

対応するステータスレジスタのビットをクリアする。

【0039】(4) 割り込み許可レジスタ

対応するステータスレジスタの各ビットによってCPU10に割り込みを発生させることを指定する。

【0040】(5) レンダリングモード

MWXは、画面の横幅が512画素以下であるか、それとも513画素以上1024画素以下であるかを指定する。GBMは1画素が8ビットであるか16ビットであるかを指定する。

【0041】MEMはグラフィックスメモリ40の種類によりアドレス不一致検出2512、2087で比較するアドレスのビット幅を指定する。

【0042】(6) 表示モード

SCMは、表示がインタレースであるか、ノンインタレースであるかを指定する。TVMは、TV同期モードであるかそれともマスタモードであるかを指定する。RCYNは、グラフィックスメモリ40のリフレッシュサイクル数を指定する。

【0043】(7) 表示サイズ

表示画面のX方向とY方向の大きさを指定する。

【0044】(8) 表示開始アドレス

グラフィックスメモリ40上の2つのフレームバッファの開始アドレスを指定する。

【0045】(9) ディスプレイリストアドレス

グラフィックスメモリ40上のディスプレイリストのスタートアドレスを指定する。

【0046】(10) ソース領域開始アドレス

テクスチャデータの格納領域の開始アドレスを指定する。

【0047】(11) 表示制御関係レジスタ

レジスタ番号10から19は、表示制御に関するレジス

タである。表示画面の大きさ等に合わせて表示データを読み出すタイミングの設定や、水平/垂直同期信号の周期等を設定する。また、表示リセット時出力レジスタは、表示読み出しを行っていない時に画面に表示するカラー値を設定する。例えば、表示動作を停止中は画面をブルーバック（青色表示）にすることができる。

【0048】(12) コマンドステータスレジスタ
ディスプレイリストのフェッチを停止した時のメモリアドレスを通知するレジスタである。

【0049】次に、CPU10がグラフィックスメモリ40をアクセスするためのFIFO方式について説明する。図5は、CPU FIFO25のブロック図である。CPU10がグラフィックスメモリ40へのストア動作を行う毎に、CPU I/F部21からライトリクエスト信号が来る。するとカウンタ252がカウントアップされるとともに、その時のCPU10のライトアドレスとデータはFIFO250に格納される。FIFO250は、16ワードのデータを蓄える。やがてカウンタ252がFIFO容量（16ワード）と比較されFIFOが満杯になったことがわかると、フリップフロップ258をセットする。その結果CPU I/F部21にはFIFOがビジーであることが通知されCPU10がこれ以上データをストアしないようにする。一方、メモリコントローラ30には、グラフィックスメモリ40への書き込み要求を出力する。メモリコントローラ30は1ワードのデータを書き込む毎にカウンタ256を更新するためのFIFOカウンタ更新信号を出力する。この時、グラフィックスメモリ40へ書き込むアドレスをレジスタ251に記憶させておき、次に書き込むアドレスとレジスタ251に記憶されているアドレスの上位複数ビットを不一致検出器2512によって比較する。この時比較されるビット幅はシステム制御レジスタ32のMEMビット320で指定される。この2つのアドレスが不一致であると（即ち、グラフィックスメモリ40に書き込むアドレスが不連続であると）フリップフロップ258をリセットする。不連続アドレスによりフリップフロップ258をリセットした場合、まだFIFOに残っているデータをグラフィックスメモリ40に書き込む為に、リセットする前までのデータの書き込みが終了したことを示す書き込み終了信号で再びフリップフロップ258をセットする。カウンタ256は一致検出器255によってカウンタ252の値と常に比較される。カウンタ256はFIFOの読み出しカウンタであり、カウンタ252はFIFOの書き込みカウンタである。この2つが一致すると（即ち、CPU10によって書き込まれたワード数だけ、メモリコントローラ30へ読み出すと）グラフィックスメモリ40への書き込みを停止するためフリップフロップ258をリセットする。また、フリーランカウンタ254は、一定期間CPU10による書き込みがなかった場合はFIFO250のデータをグラフィックス

メモリ 40 に書き込むように動作する。また、CPU 10 がグラフィックスメモリ 40 をリードする場合や、描画ユニットがディスプレイリストのフェッチを開始する場合は、これらに先駆けて FIFO250 のデータをグラフィックスメモリ 40 に書き込むように動作する。FIFO 250 は最大 16 ワードのデータを保持するので、CPU 10 からのデータ書き込みは一度に最大 16 ワード書き込みができることになる。

【0050】次に、描画用のキャッシュについて説明する。図 6 は、キャッシュ 328 のブロック図である。このキャッシュは描画専用であるが、描画ユニットはこのキャッシュ内のデータを読むことはしない。つまり、描画先の下絵とのデータ演算を行う機能を持っていないので書き込み動作のみで良い。従って、下絵をリードする必要がないのでメモリアクセス量を極端に低減し高速動作が可能となる。描画ユニット 23 がデータを書き込むとレジスタファイル 2800 に描画アドレスと描画データが記憶され、カウンタ 2801 がカウントアップされる。カウンタ 2801 がレジスタファイル 2800 が満杯になったことを検出すると、メモリコントローラ 300 に書き込みリクエストを出力する。ここでも CPU FIFO 同様にグラフィックスメモリ 40 に書き込むアドレスの不連続性を不一致検出器 2807 によって検出し不連続の場合はフリップフロップ 2802 をリセットする。この時比較されるビット幅はシステム制御レジスタ 32 の MEM ビット 320 で指定される。描画ユニット 23 は、キャッシュ 328 に空きがある状態で 1 つの図形描画コマンドが終了した場合は、前記キャッシュのデータをフラッシュさせる機能を持つ。フラッシュ信号がアクティブになると前記キャッシュは、カウンタ 2801 が示すワード数だけグラフィックスメモリ 40 にデータを書き込む。

【0051】これら 2 つに共通する点として、書き込まれた分だけしかデータ転送を行わないことが上げられる。CPU 等で用いられる一般的なキャッシュでは、ラインサイズ単位での書き込みとなるので、書き換えを行わない部分のデータも転送する。しかしここで説明したキャッシュは書き換えを行うワード数をカウントし (図 5 のカウンタ 252 による) 無駄なデータ転送をなくしている。

【0052】図 7 は以上についてアドレスが連続している場合の動作を示している。FIFO が満杯になると書き込み動作を開始しデータをメモリに転送する。この時のライトアドレスとレジスタに記憶している 1 つ前のライトアドレスを比較し、不連続でなければそのまま書き込み終わるまで書き込みリクエストを設定しておき、FIFO に書き込んだ数だけメモリに転送し終わったら書き込みリクエストを解除する。既に述べたように連続したアドレスの場合 16 ワードのデータを約 20 サイクルで書き込める。

【0053】図 8 はアドレスが不連続の場合の動作を示している。ライトアドレスが不連続であれば不連続連続信号が設定され一度書き込みリクエストを解除する。その後書き込みリクエストをリセットする前までのデータの書き込みが終了したら FIFO の残りのデータを書き込むために再び書き込みリクエストを設定する。

【0054】図 9 は、CPU 10 のアドレスマッピングの例を示したものである。CPU 10 のソフトウェアはグラフィックスメモリ 40 を主メモリ 11 と区別することなくアクセスできる。グラフィックスメモリ領域では、フレームバッファを 2 つ設けている。グラフィックスの動画表示を行う場合は、1/60 秒単位でこの 2 つの領域を切り替えて表示を行う。描画ユニット 23 は常に表示を行っていない方のフレームバッファに描画する。このようにすることによって、描画途中が表示されないために美しい動画表示が可能となる。ディスプレイリスト領域も 2 つ設ける。描画ユニット 23 が使用する領域と CPU 10 が書き込む領域を交互に使用する。

【0055】次に、表示コントローラ 24 によるグラフィックスメモリ 40 のアクセス (以下、このアクセスを表示アクセスと呼ぶ) について説明する。図 10 は表示コントローラ 24 のブロック図である。表示コントローラ 24 は表示器 51 に対して同期信号 (HSYNC, VSYNC) と表示データを出力し、表示器 51 (例えば CRT) の画面に図形を表示させるものである。タイミング制御部 246 は、同期信号 (HSYNC, VSYNC) を生成すると共に、表示データ出力制御 245 に対して、表示バッファ 29 内のデータの出力タイミングを知らせる。表示バッファ 29 は、グラフィックスメモリ 40 の表示領域のデータの一部をバッファリングする。たとえば表示バッファ 29 が 128 ワードであれば、1 画素が 1 バイトのシステムでは 256 画素分のデータを保持していることになる。また、グラフィックスメモリ 40 から表示バッファ 29 へのデータ転送速度は、表示バッファ 29 から表示器 51 へのデータ転送速度より十分高速である場合を想定している。例えばグラフィックスメモリ 40 から表示バッファ 29 へのデータ転送は 28 MHz で動作し、表示バッファ 29 から表示器 51 へのデータ転送は 14 MHz 以下で動作するものとする。従って表示バッファ 29 への書き込みは高速で行い、表示バッファ 29 の読み出しは低速で行うことになる。こうすることによって、表示すべきデータをグラフィックスメモリ 40 から読み出すタイミングが多少前後しても、表示器 51 へのデータ転送タイミングは常に一定にすることができる。表示バッファ 29 の制御は以下のように行う。

【0056】表示データ出力制御 245 は表示ドットクロック (グラフィックスプロセッサ 20 の DCLK 端子の出力であり、表示器 51 の 1 画素単位のクロック) に合わせて、読み出しアドレスレジスタ 242 が示すアド

レスから順次表示バッファ 29 を読み出すと共に読み出しアドレスレジスタ 242 を更新する。一方グラフィックスメモリアクセス制御 240 は、グラフィックスメモリアクセストリガ信号によって起動され、連続アクセスワード数に示されるワード数分だけ、グラフィックスメモリ 40 を読み出し、その読み出されたデータを表示バッファ 29 内の書き込みアドレスレジスタ 241 が示すアドレスに蓄える働きをする。前記書き込みアドレスレジスタ 241 と前記読み出しアドレスレジスタ 242 は減算器 243 で常に減算されており、その差分値は比較器 244 で定数と比較されている。つまり、前記差分値が定数 A 値以下 (例えば 12 ワード) になるとグラフィックスメモリアクセス制御 240 がグラフィックスメモリ 40 をアクセスし、表示データを表示バッファ 29 に蓄えるように動作する。

【0057】図 11 は、以上の動作をタイムチャートで示している。表示アクセスの最初の動作は HSYNC 信号によって開始される。表示アクセスが行われると表示バッファ 29 のデータワード数が増える。ある一定のワード数がたまると表示アクセスは中断され、表示データが表示器 51 に転送されるため、除々にデータが減っていく。その後データ数が定数 A 以下になると再びアクセスリクエスト信号がメモリコントローラ 30 に出力され、表示アクセスが再び行われる。1 画面が 320 × 240 ドットの場合で 1 画面分の表示を行うためには表示アクセス 1200 回行われる。

【0058】アクセスリクエスト信号が出力されてから表示アクセスが開始されるべき時間が、表示アクセス猶予時間 (TD) である。この時間を超えても表示アクセスが開始されない場合は表示バッファ 29 のデータが空になり、表示器 51 の画面が乱れることになる。前述したようにメモリコントローラ 30 は、表示アクセスを最優先に行うが、表示アクセスのリクエストが来た時に CPU 等の他のアクセスが行われている場合は表示アクセスを待たせる。ここで表示以外のアクセス期間を T_a とすると、 $T_a < T_d$ の関係を常に維持する必要がある。従って、定数 A を求めるためには T_a の最大時間を決定する必要がある。本実施例では常に連続したアドレスでメモリにアクセスするため 1 回のアクセスは約 20 サイクルで終了する。このことから T_a を容易に決定することが可能である。また、従来は不連続なアドレスでもそ

のままメモリをアクセスしていたので最悪ケース (16 ワードすべてが不連続なアドレスの場合) を考えて表示バッファ 29 のサイズを決定していたためバッファサイズが大きくなってしまいう問題があったが、本実施例により T_a を短くすることができ表示バッファのサイズを小さくすることが可能となる。

【0059】

【発明の効果】本発明によれば、グラフィックスプロセッサがメモリアクセスを行う際は常に連続したアドレスでアクセスすることによって表示の為に必要なバッファサイズを小さくでき、小型、低価格なグラフィックスプロセッサを構成することができる。

【図面の簡単な説明】

【図 1】図形処理装置のシステム構成例を示す。

【図 2】グラフィックスプロセッサ 20 の端子機能を示す。

【図 3】グラフィックスプロセッサ 20 の描画コマンドを示す。

【図 4】グラフィックスプロセッサ 20 のレジスタ機能を示す。

【図 5】CPU FIFO25 のブロック図について示す。

【図 6】キャッシュ 328 のブロック図について示す。

【図 7】連続アドレス時の動作の例を示す。

【図 8】不連続アドレス時の動作の例を示す。

【図 9】CPU 10 のアドレスマッピングの例を示す。

【図 10】表示コントローラの内部ブロック図を示す。

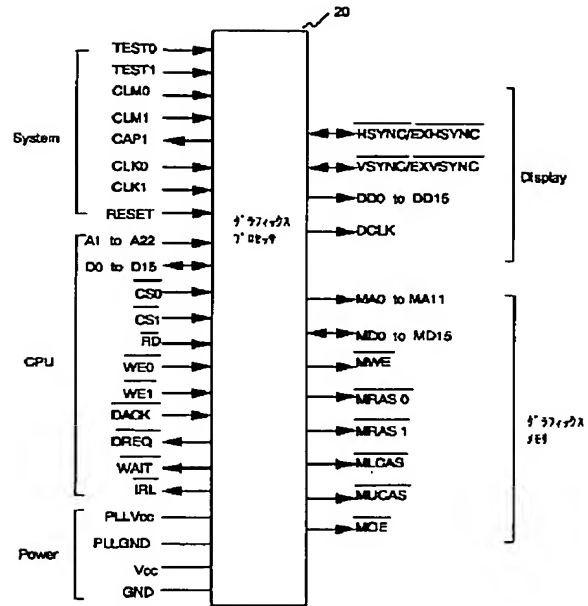
【図 11】グラフィックスメモリのアクセス内容の例を示す。

【符号の説明】

10…CPU、11…主メモリ、20…グラフィックスプロセッサ、21…CPU I/F、22…パラメータ変換部、23…描画ユニット、24…表示コントローラ、25…CPU FIFO、2512…アドレス不一致検出、26…キャッシュ 1 (コマンド用)、27…キャッシュ 2 (テクスチャ用)、28…キャッシュ 3 (描画用)、2807…アドレス不一致検出、29…表示バッファ、30…メモリコントローラ、40…グラフィックスメモリ、50…DAC (Digital to Analog Converter)、51…表示器、101…CPU 内蔵キャッシュ、320…MEMビット。

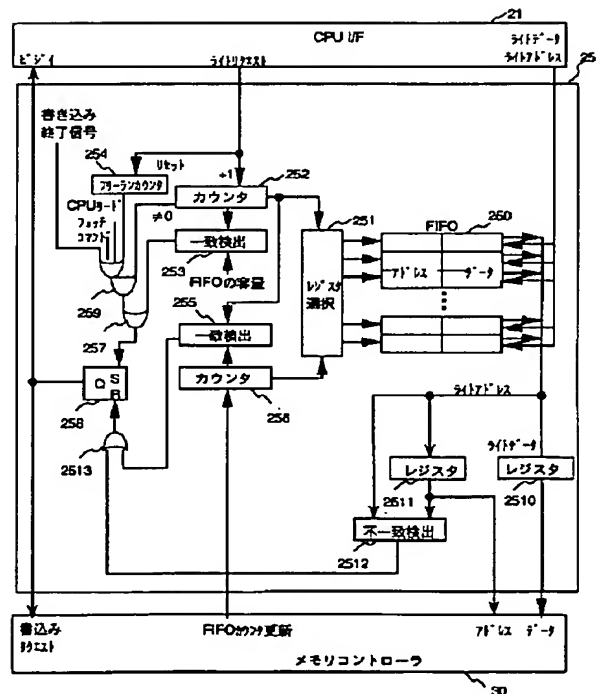
【図2】

2



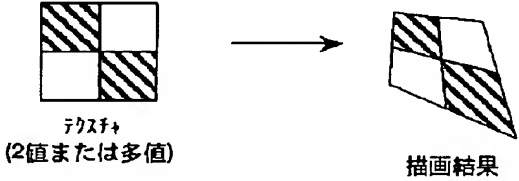

5

10



【図3】

図 3

コマンド	機能
四角形描画	<p>四角形のテクスチャパターンを変形させる</p>  <p>テクスチャ (2値または多値)</p> <p>描画結果</p>
LINE	<p>直線を描画する</p> 
MOVE	描画開始点を移動する
LOFS	ローカルオフセットパラメータを設定する
AFFIN	アフィン変換パラメータを設定する
JUMP	ディスプレイリストを分岐させる
GOSUB	サブルーチンコールする
RET	サブルーチンから復帰する
TRAP	ディスプレイリストのフェッチを終了させる
FLASH	ソースキャッシュをフラッシュする

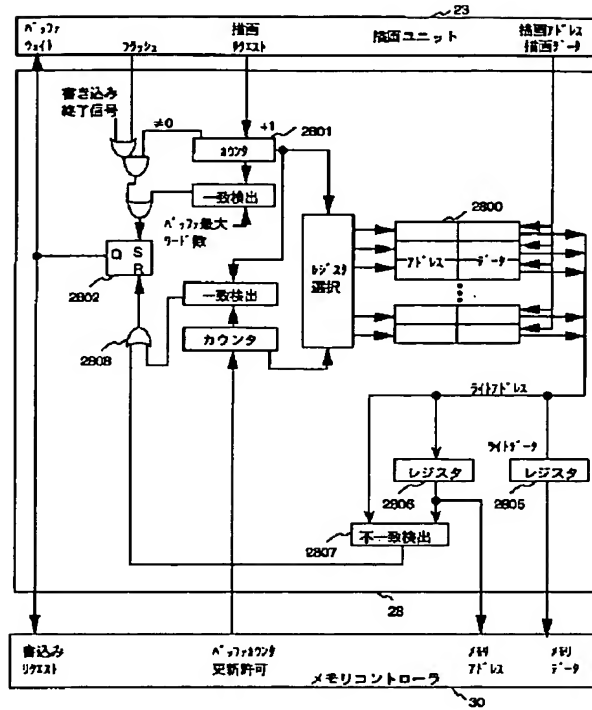
【図4】

图 4

[illegible]

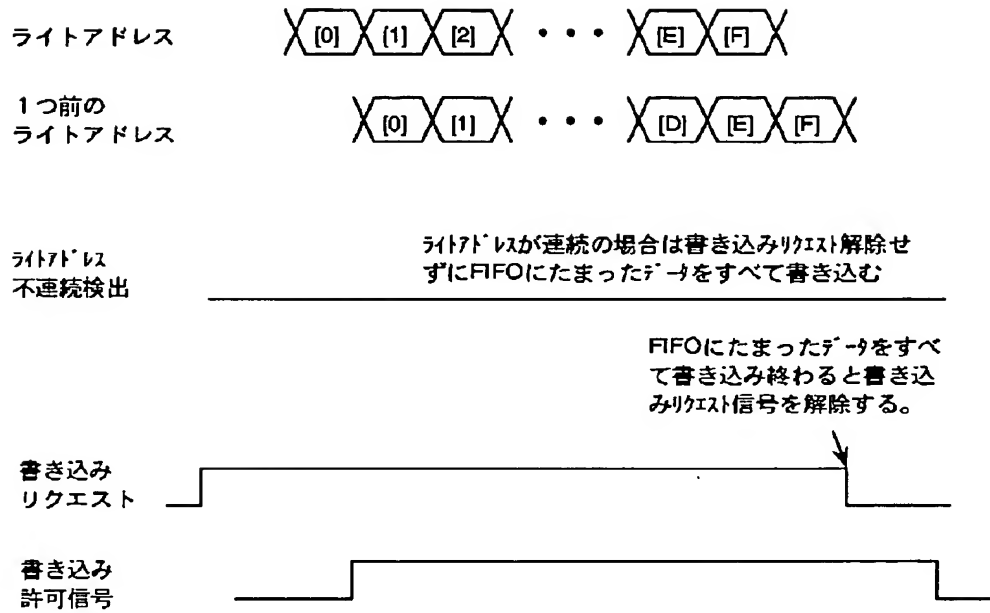
【図6】

図 6



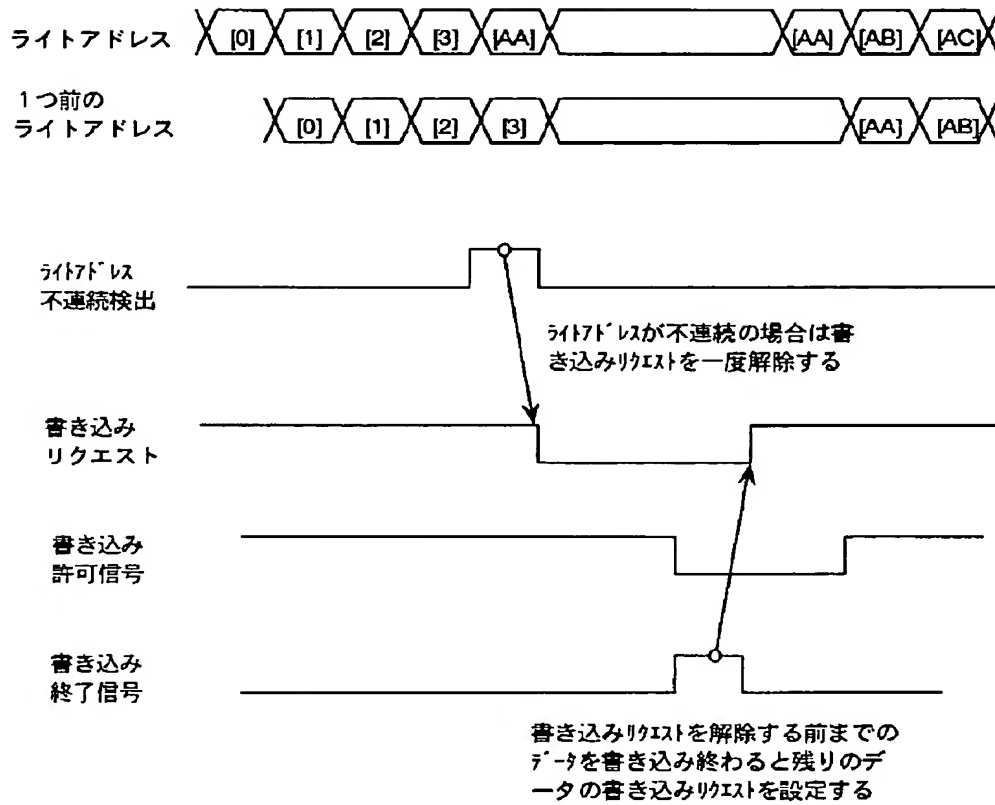
【図 7】

図 7



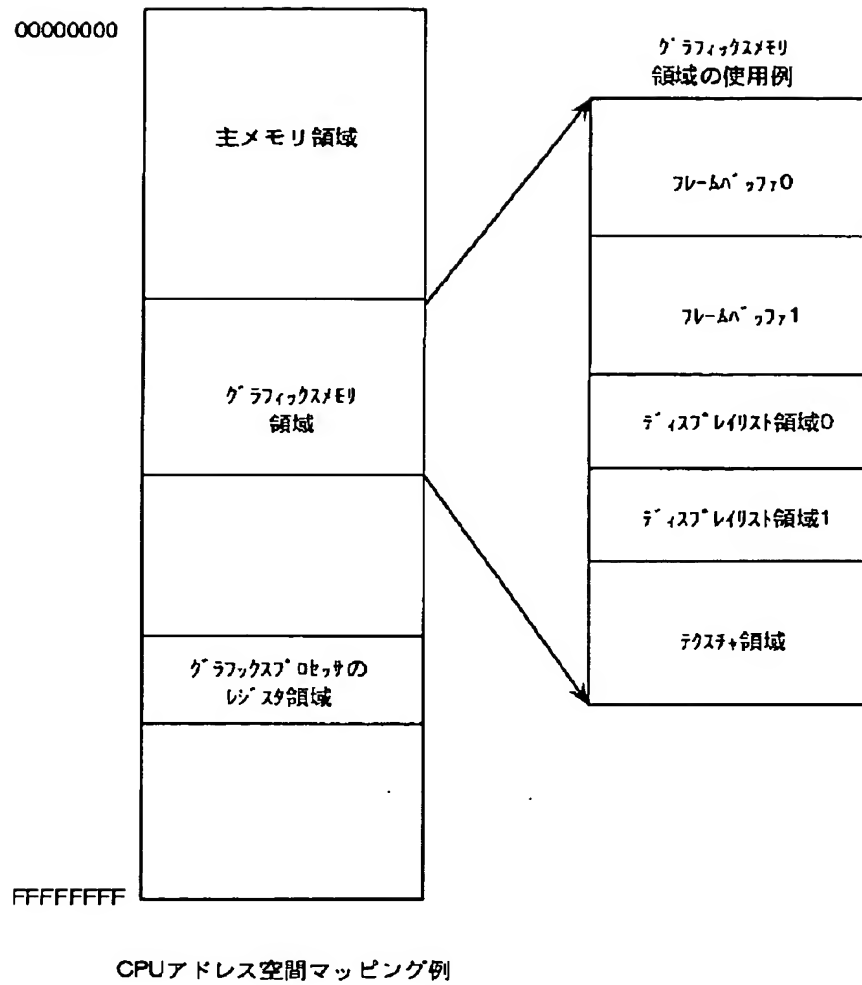
【図8】

図 8



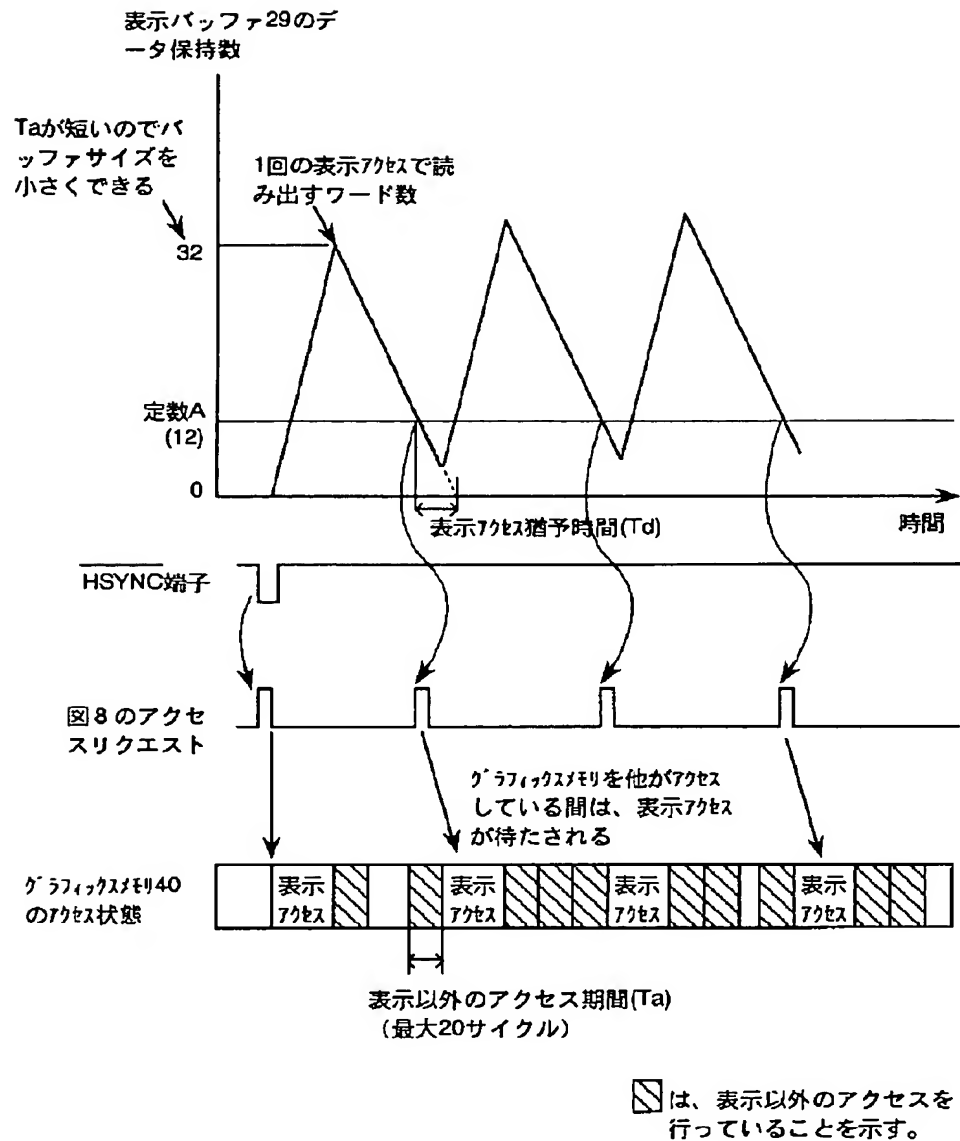
【図9】

図 9



【図11】

図 11



フロントページの続き

(51) Int. Cl. 6

G 0 6 T 11/00

G 0 9 G 5/18

識別記号

F I

G 0 9 G 5/18

G 0 6 F 15/72

A